POLITECNICO
MILANO 1863

Computer Science and Engineering

PowerEnjoy Service - Project Plan

January 15, 2017

Prof. Luca Mottola

Authors:

- ZHOU YINAN(Mat. 872686)
- ZHAO KAIXIN(Mat. 875464)
- ZHAN YUAN(Mat. 806508)

Contents

1 Introduction

1.1 Revision History

Version 1.0

1.2 Purpose and Scope

This document aims at analyzing the overall complexity and making an estimation about the project size and required effort. The result will help project manager to decide the project budget, resource allocation and the schedule of activities.The document is divided into four parts.

In the first part of the document, We will use two specific methods to estimate the size and complexity of the project. First of all, we will use Function Points to calculate the average line of codes. Secondly, we will use COCOMO method to indicate the cost and effort estimation.

In the second part of the document, we will present the tasks for the project and the corresponding schedule. We will use the above results to come up with a suitable working plan covering the entire project development.

In the third part of the document, we will assign each team member specific missions to tickle down the project.

finally, we are going to analyze the risk we may encounter during the development. By analyzing the risk and coming up with possible solutions, we'll minimize the possibility for failure.

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definitions

- Precedentedness: High if a product is similar to several previously developed projects
- Development Flexibility: High if there are no specific constraints to conform to pre-established requirements and external interface specs

- Architecture / Risk Resolution: High if we have a good risk management plan, clear definition of budget and schedule, focus on architectural definition
- Team Cohesion: High if all stakeholders are able to work in a team and share the same vision and commitment.
- Process Maturity: Refers to a well known method for assessing the maturity of a software organization, CMM, now evolved into CMMI

1.3.2 Acronyms , Abbreviations

- FP : Function Point
- ILF : Internal Logic File
- ELF : External Logic File
- EI : External Input
- EO : External Output
- EQ : External Inquires
- PREC : Precedentedness
- FLEX : Development Flexibility
- RESL : Risk Resolution
- TEAM : Team Cohesion
- PMAT : Process Maturity

1.4 Reference Documents

- Specification Document Assignments AA 2016-2017
- Function Point tables
- COCOMO tables

2 Project size, cost and effort estimation

2.1 Size estimation: function points

2.1.1 Internal Logic Files(ILFs)

2.1.2 External Logic Files(ELFs)

2.1.3 External Inputs(EIs)

2.1.4 External Inquires(EQs)

2.1.5 External Outputs(EOs)

2.1.6 Overal estinamtion

2.2 Cost and effort estimation: COCOMO II

2.2.1 Scale Drivers

2.2.2 Cost Drivers

2.2.3 Effort equations

2.2.4 Schedule estimation

3 Schedule

4 Resource allocation

5 Risk management

In this section, we'll analyze the risk we may encounter during the project development. Basically, there are three kinds of risks : Project risks, Technical risks, and Business risks. We'll present specific risks we may be trapped with and what we can do to deal with it. We use [L,M,H] to indicate the probability the risk will occur. [Low,Moderate,High]

The most possible risks come from the technical part. They threaten the quality and timeliness of the software to be produced. Some possible risks may be :

- Wrong functionality [M]: Wrong functional quality can be a serious risk and lead to meaningless workload which increases the cost and slow down the project.
  To deal with wrong functionality, we adopt a proactive risk strategy. By better writing the RASD document and increase the meeting frequency with the stakeholders, we avoid this kind of risk. Also frequently report the project process and get feedback from the stakeholders will help.

- Wrong User Interface[M] : Wrong User Interface can lead to meaningless workload.
  To deal with it, we need to frequently meet with the stakeholders and make necessary discussion.

- Bad external components[L] : Bad external components are a major issue and threat to our project. Our project largely depend on the reliability of the following external components : Google Map, DBMS, Bank Service. In case any of them fail, we need to rewrite the business logic components and this leads to significant workload and time increase.

  Although the consequence is serious, the possibility of this risk to happen is actually low. Since the external components we choose to use is from huge and reliable companies like Google and Oracle. So we adopt a reactive risk strategy here.

- Fail to accomplish logic components[L] : This risk is purely technical and there are no good solutions. Either we give programmers time to tick the task down or we recruit new employers with experience. Since the actual problem can be various, it is difficult to come up with a specific prediction. Here we adopt a reactive strategy.

The project risk may arise during the software development. These risks mainly come from the stakeholder side.

- Requirements volatility[M] : Requirement may change during the development. The point is we can not add too much constraints on our stakeholders. Here we adopt a proactive approach by organizing frequently meetings with the stakeholders. By doing this, we may avoid the risk and minimize the negative consequence.

- Unrealistic schedule/budget[M] : This risk comes from both our own side and stakeholder side. During the development we may encounter various uncertain situations and risks which will slow down our process. Also budget may be tight. Here we adopt a reactive approach because we do not know how to come up with a efficient solution until we actually get trapped by the problem.

The third part of the risk is business risk. Some possible situations are presented below:

- Management risk[L] : There is a possibility of losing the support of senior management due to a change in focus or a change in people. Since we assume there are three members in our development team, it is actually kind of critical issue, however the possibility is not high. So we adopt a reactive strategy here.

- Budget risk[L] : Losing budget or personnel commitment is not so likely in our case, so we adopt a reactive strategy here.

6 Effort